

# 長方形を利用した正面画像生成

岡本 清明 徐 剛

立命館大学 理工学研究科

あらまし CGにおいては高速化のためにテクスチャマッピングをアフィン変換を用いて行っている。そのため生成された画像のテクスチャは、大きな物体でかつ対象平面に対し角度が大きい場合歪みを生じてしまう。その歪みを極力減少させるために、それぞれに対応した実写の画像を正面から見たような画像に変換する。本論文では3次元モデルの面が多くの場合に長方形であることに着目し、長方形の画像を簡単に正面画像変換する手法を提案する。1枚の実画像中の3次元モデルの面に対応する部分の4頂点を取得し、長方形の特性を利用しカメラ座標系と世界座標系間の関係式を求める。その関係式からカメラの焦点距離、長方形の縦横の長さを求め、正面画像へ変換する射影変換行列を求める。最後に射影行列を用いて正面画像を生成する。

キーワード テクスチャマッピング, 正面画像, 焦点距離, 射影行列, 長方形

## Generating Frontal Texture Map Using Rectangles From Real Images

Kiyoaki Okamoto and Gang Xu

Graduate School of Science and Engineering, Ritsumeikan University

**Abstract** In texture mapping, to reduce deformation while keeping the efficiency of affine projection, we generate frontal texture map for each 3D plane using homography. We note that in many cases, especially in man-made environment, many faces of a 3-dimensional object are rectangles, and we propose a method of generating frontal texture map from images of rectangles. At first, we get the 4 vertexes of a rectangle in image that correspond to a face of 3-dimensional model, and determine the equation of describing the relationship between the camera coordinatesystem and the world coordinate system. And then, we determine the focal length, the aspect ratio of the rectangle, homography matrix. Finally we generate the frontal image using the obtained homography matrix.

**Key words** texture mapping, frontal image, focal length, homography, rectangle

## 1 はじめに

リアルな仮想空間を表現するためには、表面の質感、現実感を高精度に表現する3次元モデルが必要となる。そのため、実物体に忠実な3次元形状モデルを自動生成する方法の確立が期待されている。

最近では、コンピュータグラフィックス(CG)・コンピュータビジョン(CV)研究やCADなどの発展により、貴重な文化財・文化遺産、歴史豊かな文化や

芸能などのデジタル記録・保存、デジタルアーカイブ化・コンテンツ化の研究が盛んに進められている[1][2]。最近では、飛行機からレーザースキャナーを用いて、3次元点群データをスキャンしてリアルな3次元地図を作成するということが行われている。そして、このような3次元モデルに質感、現実感を与えるには、テクスチャが必要となり、テクスチャを貼るには、人からの視点による観測データを用いた方が、より現実感を与えることができ、より自然な3次元

モデルができるはずである。

CGにおいて、3次元モデリングされた物体へテクスチャマッピングする場合には、高速化のためにテクスチャをアフィン変換し貼り付けることで、マッピングが行われている [3]。テクスチャデータを作成するには、ペイントソフトで作成したり、実際の写真や映像を用いる場合の2通りの方法がある。ペイントソフトを用いた場合は、正面画像になっているため画像の歪みはあまり問題はない。しかし、ペイントソフトを利用する場合、誰もが簡単に作成できるわけではなく、デザイナー等が作成しなければ表面の質感、現実感をリアルに表現することは難しい。そこで、誰でも容易に得られる写真や映像を用いる方法について考える。実際の画像を用いてそのままテクスチャとして貼った時、大きな歪みを生じる場合がある。対象平面を斜めから撮影した写真を用いるとひどい歪みが生じる。この画像の歪みを減少させるためには、対象物体の正面から見た画像が必要である。しかし、実際にマッピングする面に合わせた真正面からの画像を入手することは一般的に不可能である。そのために、3次元モデルへ実画像をマッピングするには、その3次元モデルに対応している面を真正面から見ているような画像に合うように変換する必要がある。これが可能であるならば、任意の角度、視点から写真を撮影するだけでテクスチャを作成でき、大規模な3次元モデルへのテクスチャ作成時に大幅にも人の労力を削減することができる。

ここで画像を正面画像にするには、カメラの焦点距離や外部変数を求める必要がある。そして、これはカメラキャリブレーションを行うことと同じであり、カメラキャリブレーションはコンピュータビジョンの研究では、以前から多くの手法が提案されてきている。

そこで、本論文では、3次元モデルの面が多くの場合に長方形であることに着目し、長方形の画像を簡単に正面画像変換する手法を提案する。本論文の構成は、第2章に従来で研究されてきた手法、第3章では座標系定義、第4章では正面画像を生成するために必要なパラメータの算出方法、第5章では算出されたシミュレーションデータと実画像を用いた実験結果となっている。

## 2 従来手法

正面画像を生成するための必要なパラメータとして、焦点距離、回転行列、並進ベクトルが挙げられる。これらのパラメータの推定は以前から研究され、カメラキャリブレーションもそれにあたり、以下にいくつかの研究されてきたキャリブレーション手法について述べる。

消失点を用いた手法 [4]。3次元空間で平行線である場合、画像平面へ射影されると1点で交わる。この点は消失点 (FOE) と呼ばれている。そして、カメラの焦点から消失点に延びる直線が直行することを利用して、方程式を立てることができる。そしてその方程式を解くことで、焦点距離、回転行列、並進ベクトル

を求めていく。この手法を用いると画像1枚のみで実空間で平行な直線2本が画像に射影されて、画像内に存在すれば求めることが可能である。しかし、画像上に写っている直線の2本が平行に近い場合、算出される焦点距離が不安定となり、回転行列、並進ベクトルの値が求まらない場合や正確さを失う場合がある [5]。次に平面を利用したキャリブレーションについて述べる。異なった角度から撮影された平面の画像を5枚用いた手法 [6]。初期値をセットすることが難しく、異なった視点から撮影された画像が5枚は最低必要である。モデル平面と画像間の射影行列を用いた手法 [7]。モデル平面の世界座標系のZ座標を0とし、3章で記述されている方程式を変形していく。そしてそこで得られた方程式と回転ベクトルの要素が直交であることを利用し2つの式を得ることができる。これは射影行列が8の自由度を持ち、2つの内部変数と6つの外部変数を持っているからである。そして、この2つの式を最尤推定法に基づいた非線形の最適化手法を用いて解いていく。

次に、正面画像生成方法の従来手法を記述する。まず、3次元形状が既知のモデルと撮影した写真との間の関係を定める射影行列を求める。そして、求めた射影行列からカメラ座標系と物体の座標系との回転行列と並進ベクトルを求める。更に、それを利用して各パッチにおける射影行列を求め、物体の面にあわせた正面画像を生成する手法 [8]。これは、3次元モデルの座標が既知でなければならないという前提条件が必要であった。

従来手法においてはカメラの外部変数や焦点距離を求める上で、複数の画像が必要であったり、対象平面の寸法が事前にわかっていることが必要であった。そこで本手法では、対象平面を長方形に限定したより簡便な手法を提案する。

## 3 座標系定義

普段私達が見る画像はデジタル画像と呼ばれるものであり、デジタル画像座標は座標の原点は左上にあり、右方向が横軸  $u$ 、下方向が縦軸  $v$  である。しかし、射影関係を記述する場合には、正規化画像座標系を用いる。正規化画像とは焦点から単位長のあるところにある画像である。そこで、デジタル画像座標を正規化画像座標へ変換する。式は以下のようになる。ここでは拡張ベクトルを示す。

$$\hat{\mathbf{n}} = \mathbf{P}\hat{\mathbf{x}} \quad (1)$$

ここで、 $\hat{\mathbf{n}} = [u, v, 1]^T$ 、 $\hat{\mathbf{x}} = [x, y, 1]^T$  はそれぞれデジタル画像座標  $(u, v)$  と正規化画像座標  $(x, y)$  の拡張ベクトルである。行列  $\mathbf{P}$  はカメラ内部行列であり、以下のように表される [9]。  $[u_0, v_0]$  は画像中心の座標である。

$$\mathbf{P} = \begin{bmatrix} \alpha_u & b & u_0 \\ 0 & \alpha_v & v_0 \\ 0 & 0 & 1 \end{bmatrix}$$

しかし、最近のカメラの性能の向上により、行列Pは以下のようにも表されるようになった。本論文でもこれを利用する。

$$P = \begin{bmatrix} f & 0 & u_0 \\ 0 & f & v_0 \\ 0 & 0 & 1 \end{bmatrix}$$

次に、正規化座標系を世界座標系に変換する。これは正規化座標系における座標を  $M_c$ 、世界座標系における座標を  $M_w$ 、回転行列  $R$ 、並進ベクトルを  $t$  とすると、以下の式のように表すことができる。イメージ図を図1に示す。

$$M_c = RM_w + t \quad (2)$$

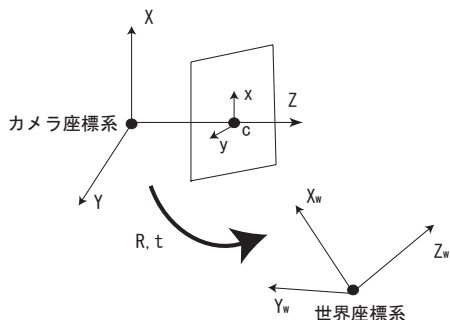


図 1: 座標系の関係のイメージ図

## 4 正面画像生成

### 4.1 長方形の3次元モデルと画像間の射影関係

3次元モデルの側面である長方形が画像に射影される様子を図2に示す。

図3のように世界座標系において、3次元モデルの側面をXY平面にしZ座標を0にする。そして面の中心が(0,0,0)になるように定義する。ここで、図中の  $a, b$  はそれぞれ長方形の縦、横の長さの半分である。

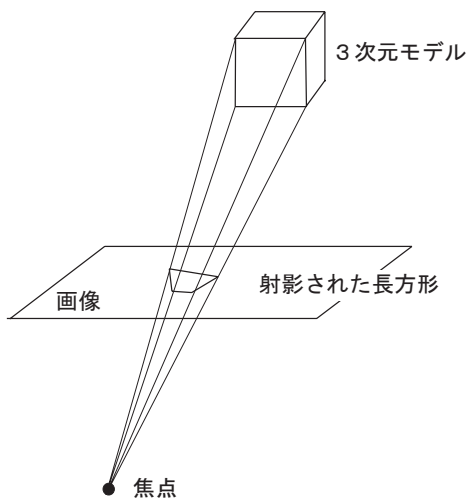


図 2: 長方形が射影される様子

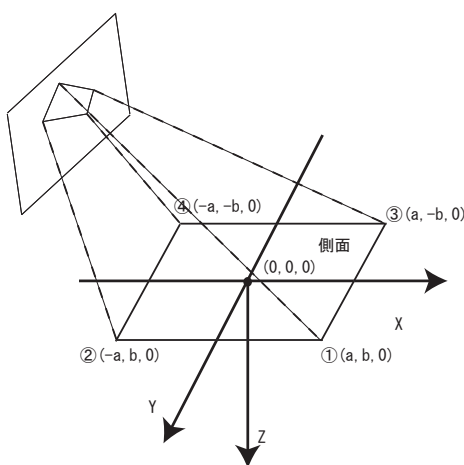


図 3: 実画像と3次元モデルの関係のイメージ図

すると、式(1)、式(2)より式(3)のように表すことができる。ここで、回転行列は  $R = [r_1, r_2, r_3]$ 、並進ベクトルは  $t$  とする。

$$\begin{bmatrix} X_c \\ Y_c \\ Z_c \end{bmatrix} = R \begin{bmatrix} X \\ Y \\ 0 \end{bmatrix} + t$$

$$\tilde{\mathbf{m}} = \begin{bmatrix} u - u_0 \\ v - v_0 \\ 1 \end{bmatrix} = \begin{bmatrix} f & 0 & 0 \\ 0 & f & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X \\ Y \\ 1 \end{bmatrix} \quad (3)$$

よって、各頂点に対しての関係式は以下の4式となる。 $s$  は未知のスケール、 $H = [r_1, r_2, t]$  とし、

$$\mathbf{A} = \begin{pmatrix} f & 0 & 0 \\ 0 & f & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

とする。ここでそれぞれの式で表されている数式記号中の番号は、図 3 で示した頂点番号と対応している。

$$s_1 \tilde{\mathbf{m}}_1 = \mathbf{A}\mathbf{H} \begin{bmatrix} a \\ b \\ 1 \end{bmatrix} \quad (4)$$

$$s_2 \tilde{\mathbf{m}}_2 = \mathbf{A}\mathbf{H} \begin{bmatrix} -a \\ b \\ 1 \end{bmatrix} \quad (5)$$

$$s_3 \tilde{\mathbf{m}}_3 = \mathbf{A}\mathbf{H} \begin{bmatrix} a \\ -b \\ 1 \end{bmatrix} \quad (6)$$

$$s_4 \tilde{\mathbf{m}}_4 = \mathbf{A}\mathbf{H} \begin{bmatrix} -a \\ -b \\ 1 \end{bmatrix} \quad (7)$$

式 (4), 式 (5), 式 (6), 式 (7) より,

$$s_1 \tilde{\mathbf{m}}_1 + s_4 \tilde{\mathbf{m}}_4 = s_2 \tilde{\mathbf{m}}_2 + s_3 \tilde{\mathbf{m}}_3$$

となり, これを変形すると,

$$\begin{bmatrix} \tilde{\mathbf{m}}_1 & -\tilde{\mathbf{m}}_2 & -\tilde{\mathbf{m}}_3 & \tilde{\mathbf{m}}_4 \end{bmatrix} \begin{bmatrix} s_1 \\ s_2 \\ s_3 \\ s_4 \end{bmatrix} = 0$$

となる。これより,

$\mathbf{B} = \begin{bmatrix} \tilde{\mathbf{m}}_1 & -\tilde{\mathbf{m}}_2 & -\tilde{\mathbf{m}}_3 & \tilde{\mathbf{m}}_4 \end{bmatrix}$  とすると,  $\mathbf{B}^T \mathbf{B}$  の最小固有値に対応する固有ベクトルとして,  $s_1, s_2, s_3, s_4$  を求めることができる。空間の尺度は本来確定できないが,  $s_1, s_2, s_3, s_4$  の自乗和が 1 であることにより, とりあえず与えられる。しかし, ここで対応している 3 次元モデルに対して対応しているテクスチャが正面画像に近い場合, 求まった固有ベクトルが求まらない場合がある。これは, 固有ベクトルを求める際に起こる退化が原因である。この場合, 固有ベクトルが不安定となるため, この後のパラメータを算出することができない。したがって, 固有ベクトルが求まっているかどうかを判断する必要がある。ここで, 固有ベクトルの 3 番目に注目する。正し

い値が求まっている場合, 行列のランクは 3 であるが, ランクが 2 であれば, 不安定と判断する。

次に, 式 (4), 式 (5), 式 (6), 式 (7) を変形すると, 以下の 4 式となる。

$$\mathbf{A}^{-1} s_1 \tilde{\mathbf{m}}_1 = \mathbf{H} \begin{bmatrix} a \\ b \\ 1 \end{bmatrix} \quad (8)$$

$$\mathbf{A}^{-1} s_2 \tilde{\mathbf{m}}_2 = \mathbf{H} \begin{bmatrix} -a \\ b \\ 1 \end{bmatrix} \quad (9)$$

$$\mathbf{A}^{-1} s_3 \tilde{\mathbf{m}}_3 = \mathbf{H} \begin{bmatrix} a \\ -b \\ 1 \end{bmatrix} \quad (10)$$

$$\mathbf{A}^{-1} s_4 \tilde{\mathbf{m}}_4 = \mathbf{H} \begin{bmatrix} -a \\ -b \\ 1 \end{bmatrix} \quad (11)$$

式 (9) と式 (11), 式 (10) と式 (11) との差, 式 (9) と式 (10) の和より, 以下の 3 式が得られる。

$$\mathbf{A}^{-1}(s_2 \tilde{\mathbf{m}}_2 - s_4 \tilde{\mathbf{m}}_4) = 2br_2 \quad (12)$$

$$\mathbf{A}^{-1}(s_3 \tilde{\mathbf{m}}_3 - s_4 \tilde{\mathbf{m}}_4) = 2ar_1 \quad (13)$$

$$\mathbf{A}^{-1}(s_2 \tilde{\mathbf{m}}_2 + s_3 \tilde{\mathbf{m}}_3) = 2t \quad (14)$$

次に, 式 (12) と式 (13), 式 (12), 式 (13) の内積をとる。

$$(s_2 \tilde{\mathbf{m}}_2 - s_4 \tilde{\mathbf{m}}_4)^T \mathbf{A}^{-T} \mathbf{A}^{-1}(s_3 \tilde{\mathbf{m}}_3 - s_4 \tilde{\mathbf{m}}_4) = 0 \quad (15)$$

$$(s_2 \tilde{\mathbf{m}}_2 - s_4 \tilde{\mathbf{m}}_4)^T \mathbf{A}^{-T} \mathbf{A}^{-1}(s_2 \tilde{\mathbf{m}}_2 - s_4 \tilde{\mathbf{m}}_4) = 4b^2 \quad (16)$$

$$(s_3 \tilde{\mathbf{m}}_3 - s_4 \tilde{\mathbf{m}}_4)^T \mathbf{A}^{-T} \mathbf{A}^{-1}(s_3 \tilde{\mathbf{m}}_3 - s_4 \tilde{\mathbf{m}}_4) = 4a^2 \quad (17)$$

$$\begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix} = (s_2 \tilde{\mathbf{m}}_2 - s_4 \tilde{\mathbf{m}}_4)$$

$$\begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix} = (s_3 \tilde{\mathbf{m}}_3 - s_4 \tilde{\mathbf{m}}_4)$$

とすると、式 (15) は、以下のように表され焦点距離  $f$  が求まる。

$$\begin{aligned} & (s_2\tilde{\mathbf{m}}_2 - s_4\tilde{\mathbf{m}}_4)^T \mathbf{A}^{-T} \mathbf{A}^{-1} (s_3\tilde{\mathbf{m}}_3 - s_4\tilde{\mathbf{m}}_4) \\ &= \frac{1}{f^2} (u_1v_1 + u_2v_2 + u_3v_3 f^2) = 0 \end{aligned}$$

これを  $f$  について解くと以下ようになる。

$$f = \sqrt{-\frac{u_1v_1 + u_2v_2}{u_3v_3}} \quad (18)$$

この求まった焦点距離  $f$  を、式 (16)、式 (17) に代入すると、長方形の縦横の長さの半分  $a, b$  が求まる。更に、求まった焦点距離  $f$ 、長方形の縦横の長さの半分  $a, b$  を式 (12)、(13)、(14) に代入すると、行列  $\mathbf{H}=[\mathbf{r}_1\mathbf{r}_2\mathbf{t}]$  を求めることができる。更に、 $\mathbf{r}_3 = \mathbf{r}_1 \times \mathbf{r}_2$  として回転行列と並進ベクトルに分解できる。

## 4.2 アルゴリズム

本手法のアルゴリズムについて述べる。ここで、正面画像生成にあたって必要なパラメータの数として、焦点距離、回転行列、並進ベクトル、長方形の縦横の長さの比の計 8 つである。式 (4)、式 (5)、式 (6)、式 (7) より、8 個の方程式があるので、正面画像生成に必要なパラメータを求めることがわかる。図 4 にアルゴリズムを示す。

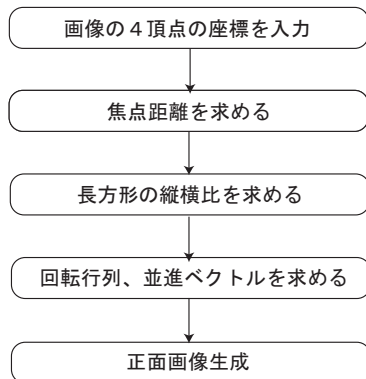


図 4: アルゴリズム

## 5 実験

### 5.1 シミュレーション

シミュレーション実験をした。まず、真値から画像上の 4 点の座標を算出しそれに対して、ガウスノイズを加えた値を入力とし実験をした。表 1 に真値と実験結果を示す。表内の値は、小数点第 4 位を四捨五入した値である。

表 1: シミュレーション実験結果

	真値	=0.0	=1.0	=2.0	=3.0
$f$	1000	1000	999.485	999.271	999.107
a に対する b の長さ	2.000	2.000	2.000	2.001	2.001
回転行列	-0.695	-0.695	-0.695	-0.695	-0.695
	0.714	0.714	0.714	0.714	0.714
	0.089	0.089	0.089	0.089	0.089
	-0.192	-0.192	-0.193	-0.193	-0.193
	-0.304	-0.304	-0.304	-0.304	-0.304
	0.933	0.933	0.933	0.933	0.933
	0.693	0.693	0.693	0.693	0.693
	0.631	0.631	0.631	0.631	0.631
並進ベクトル	0.348	0.348	0.349	0.349	0.349
	100	0.344	0.344	0.344	0.345
	200	0.688	0.688	0.688	0.688
	20	0.069	0.069	0.069	0.069

ここで、並進ベクトルの値が異なるのは、スケールが任意のためであり、並進ベクトルの比が重要となる。すべての結果を見ると、本手法の有効性が正しいということが証明された。

### 5.2 実データを用いた実験

実データを用いた実験として、デジタルカメラで撮影した画像を用いた。まず、図 5 のアーチェリー型となっている部分の長方形の 4 頂点の座標を入力し、正面画像を生成した。結果を図 6 に示す。



図 5: 原画像 1



図 6: 原画像 1 を正面画像へ変換した結果

次に、立命館大学琵琶湖草津キャンパス内にある建物を複数の角度から撮影した画像を用い、3次元復元したものをVRML上に定義し、図5を正面画像に変換した図6を3次元モデルの側面に貼り付けて視覚的な比較を行った。結果を図7、図8に示す。



図 7: 原画像 1 を貼り付けた結果



図 8: 正面画像を貼り付けた結果

視覚的に比べると正面画像に変換せずに貼り付けた場合、大きな歪みが生じているのがわかる。特に画像内の線に注目すると、実際では、直線であるのに対し、曲線になっているのがわかる。それに対し正面画像に変換してから貼り付けた場合、直線部分が忠実に表現されていることがわかる。以上から、正面画像に変更することにより歪みが減少できたことがわかる。

## 6 むすび

本論文では、3次元モデルの多くが長方形であることに着目し、長方形を利用した正面画像生成手法を提案した。更に、正面画像に変換することにより、テクスチャマッピングする際に生じる歪みを減少させることを示した。本手法では、3次元モデルの貼り付ける部分が長方形という条件を満たせば、実画像

データ中の長方形の4頂点の座標を入力することで、正面画像を生成することができるというものであった。本手法の有効性を証明するために、シミュレーションと実画像データによる実験を行った。シミュレーションについては、小数点以下で若干の誤差はあるが、ほぼシミュレーション値と同じ値を得ることができた。実画像データによる実験では、原画像を変換せず貼り付けた結果と正面画像に変換し貼り付けた結果を比較すると、原画像をそのまま貼り付けた場合には、画像内に歪みが生じているが、画像を貼り付けた場合は、あまり歪みが生じなかったことにより、本手法の有効性が確認できた。

## 謝辞

本研究は文部科学省21世紀COEプログラム「京都アート・エンタテインメント創成研究」の助成を一部受けた。

## 参考文献

- [1] Asanobu KITAMOTO, Sonoko SATO, Takeo YAMAMOTO, and Kinji ONO "Context Recombination for Digital Cultural Archives", Proceedings of the International Conference on Digital Archive Technologies (ICDAT2004), pp.105-119, May.2004.
- [2] 早乙女恵子, 高橋季穂, 笹田晋司, 佐藤誠"インタラクティブ性を持つニューアーカイブの研究", 芸術科学会論文誌, Vol.1 No.3 pp105-107
- [3] 杉原厚吉, "グラフィックスの数理", 共立出版, 1996
- [4] B. Caprile and V. Torre, "Using Vanishing Points for Camera Calibration", The International Journal of Computer Vision,4(2):127-140,Mar.1990.
- [5] Virginio Cantoni, Luca Lombardi, Marco Porta, Nicolas Sicard "Vanishing Point Detection:Representation Analysis and New Approaches", Proceedings of the 11th International Conference on Image Analysis and Processing(ICIAP 2001).
- [6] B. Triggs, "Autocalibration from planar scenes", In Proc. 5th European Conference on Computer Vision, pages 89-105, Freiburg, Germany, June 1998.
- [7] Zhengyou Zhang, "Flexible Camera Calibration By Viewing a Plane From Unknown Orientations", ICCV 1999: 666-673.
- [8] 松井 祐司, 徐 剛, "実画像を用いた3次元物体への正面テクスチャマップの作成", 情報処理学会研究報告, Vol99, No.3, p25-p32, 1999
- [9] 徐 剛, 辻 三郎, "3次元ビジョン", 共立出版, 1998